迭代自组织哈希算法 *

韩雪莲，田爱奎+，王　振，卢海涛

(山东理工大学 计算机科学与技术学院, 山东 淄博 255000)

摘　要：为了解决现有哈希算法的中心点不确性和离散编码的表达有限的问题，提出迭代自组织哈希算法(iterative self-organizing hashing，ISOH)。该算法采用迭代自组织数据分析量化空间，以提高近邻检索准确率；在聚类中心初始化方面，使用最远平均距离方法选择初始聚类中心，避免初始聚类中心的随机性；为解决固定编码长度所表示的二值编码种类有限的问题，提出建立多重编码机制；在时间复杂度方面，ISOH 算法采用乘积空间，以较低的代价得到更长的编码。实验结果表明，在 SIFT、GIST 和 CIFAR10 数据集上与 K 均值哈希和可扩展图哈希等具体化哈希算法相比，ISOH 算法能有效提高近邻检索的准确率。

关键词：迭代自组织数据分析；多重编码；乘积空间；最远平均距离

中图分类号：TP391.41　　doi: 10.19734/j.issn.1001-3695.2018.10.0811

Iterative self-organizing hashing algorithm

Han Xuelian, Tian Aikui+, Wang Zhen, Lu Haitao

(*College of Computer Science & Technology*, *Shandong University of Technology*, *Zibo Shandong* 255000, *China*)

Abstract: To fix the randomness of the cluster centers and the limited representation of the discrete binary codes, this paper presented a method termed Iterative Self-organizing Hashing (ISOH) . This algorithm employed the Iterative Self-organizing Data Analysis to quantify the original space. As a result, the above measurement improves the retrieval accuracy largely. During initializing the clustering centers, this method utilized the farthest average distance to fix the randomness problem. As the fixed binary bits can represent a limited number of the codes, the hash based image ANN retrieval method has poor performance. To this end, this paper established the multi-encoding mechanism. In terms of the training time complexity, this method employed the product space mechanism to obtain longer encoding results at a lower cost. This paper conducted the comparative experiments in SIFT, GIST and CIFAR10 datasets. The experimental results show that ISOH is superior K-means Hashing and Scalable Graph Hashing etc. in achieving image ANN retrieval.

Key words: iterative self-organizing data analysis; multiple coding; product space; farthest average distance

0　引言

随着互联网技术应用的成熟，图像、视频等数据呈现爆炸式增长，如何在海量数据中快速找到人们感兴趣的图像已成为研究热点。早在 20 世纪 70 年代人们就已经提出基于文本的图像检索技术[1,2]，采用人工标注图像的方法检索相似图像，操作简单，检索速度较快。但随着图像规模不断扩大，人工标注图像变得越来越困难。同时，由于文字描述不能确切地表达图像的语义信息，导致某些检索结果不符合用户的需求，如必应、百度和 360 搜索引擎上搜索关键词“篮球”，结果如图 1 所示。从图中可以看出，返回图像中除篮球外，还有篮球明星、篮球筐和篮球场地等。为了解决基于文本的图像检索技术的不足，学者们提出了基于树结构的图像检索技术[3]。

基于树结构的图像检索技术[3]以树结构存储图像特征，并为每个叶子节点设定一个阈值。在检索近邻图像时，利用每一层的树型结构，快速剔除大部分数据来提高近邻检索速度。以 K-D[4]树为例，其工作方式(图 2)因形似大树而得名，通过对查询空间的不断细分，并对细分后的空间进行同时查询，从而达到加快检索速度的目的。但是随着特征维度的增加，基于树型索引结构的检索效率将变低。为解决这一问题，学者们引入了基于哈希的图像检索技术[5]。

基于哈希的图像检索技术[5]的基本思想是将高维浮点向量表示成紧凑二进制编码，并根据汉明距离检索近邻点。最早的哈希算法是局部敏感哈希算法(locality sensitive hashing, LSH)[6]，其随机生成线性哈希映射函数，并根据数据点与线性哈希函数的映射结果生成二进制编码。LSH 算法的哈希函数是随机生成的，对训练数据的依赖性较弱，需要生成相对较长的二进制编码才能产生较好的近邻检索效果。为了保证采用紧凑二进制编码也能得到较优的近邻检索结果，Shen 和 Weiss 等人[7,8]提出谱哈希算法(spectral hashing, SH)，通过分割谱图来学习数据点的二进制编码。其图形建模的复杂度高，而且要求数据集服从均匀分布，可是实际数据集并不符合这一要求。针对图形建模复杂度高的问题，Jiang 等人[9]提出了可扩展图哈希算法(scalable graph hashing, SGH)。该算法可以通过特征变换方式有效地逼近整个图，无须再显式计算成对的相似图矩阵。但是在实际应用中，SGH 算法在进行特征变换时，参数 ρ 需要通过交叉验证技术来调整。与基于图的哈希算法不同，主成分分析哈希算法(principal component analysis hashing, PCAH)[10]和随机旋转哈希算法(random

收稿日期：2018-10-24；**修回日期：**2018-12-24　**基金项目：**山东省自然科学基金资助项目（ZR2018PF005）

作者简介：韩雪莲（1994-），女，山东烟台人，硕士研究生，主要研究方向为计算机视觉；田爱奎（1964-），男（通信作者），山东莱芜人，教授，硕导，博士，主要研究方向为计算机视觉、机器学习（takui@sdut.edu.cn）；王振（1988-），男，山东枣庄人，讲师，博士，主要研究方向为计算机视觉；卢海涛（1994-），男，河南周口人，硕士研究生，主要研究方向为计算机视觉.

rotating hashing, RR)[11,12]采用超立方体量化和编码浮点数据，但超立方体顶点是固定的，灵活性差，对数据点集的空间分布适应能力弱。K 均值哈希算法(K-means hashing, KMH)[5,13,14]通过 K-means 对数据集聚类，数据间的距离用相对应的聚类中心间的距离近似，达到允许超立方体进行拉伸的效果，具有较好的灵活性，但是 K-means 算法的初始聚类中心的选择是任意的，聚类效果不稳定，而且聚类中心数在聚类过程中是固定不变的，导致 KMH 算法的近邻检索性能不佳。



图 1　搜索文本“篮球”的检索结果

Fig. 1　Retrieval results of search text "basketball"

图 2　K-D 树结构

Fig. 2　K-D tree structure

为解决上文中提到的几种算法存在的灵活性差、准确率低等问题，本文提出了迭代自组织哈希算法(ISOH)，采用迭代自组织数据分析(iterative self-organizing data analysis, ISODATA)[15]、最远平均距离方法、多重编码[16]和乘积空间[5,17]等技术，使算法具有高准确率。其算法流程如图 3 所示。首先，通过乘积空间对训练数据集和测试数据集进行预处理(图 3(a))；其次，利用相似性保持方法获得聚类中心 Z，并根据聚类中心对训练数据集(预处理后)和测试数据集(预处理后)进行归类操作(图 3(b))；然后，建立多重编码机制对其进行编码(图 3(c))；最后，计算汉明距离 $h(X_{it},Y_{jt})$(图 3(d))。

本文算法具有如下创新点：

a) 在空间量化方面，ISOH 算法使用了迭代自组织数据分析[16]算法。与 KMH 算法相比，ISODATA 算法可根据各个类所包含数据点的实际情况，动态地调整聚类中心的数目，以达到较好地编码效果。

b) 为解决初始聚类中心的随机性和盲目性的问题，本文使用最远平均距离方法选择初始聚类中心。最远平均距离方法先将全体样本点作为一类，其平均值作为第一个聚类中心点，距离第一个聚类中心点最远样本点为第二个聚类中心点，之后依次对类中样本点数多者进行分裂处理，直到聚类中心数到达给定数目为止。

c) 本文建立多重编码[16]机制，有效解决了基于哈希的图像检索技术中由固定编码长度导致表示种类有限的问题。

d) 为了能以较低的代价得到更长的编码，ISOH 算法运用乘积空间[5,17]，通过对原始空间进行平衡划分，再对每个子空间进行相应操作来降低算法的时间复杂度。

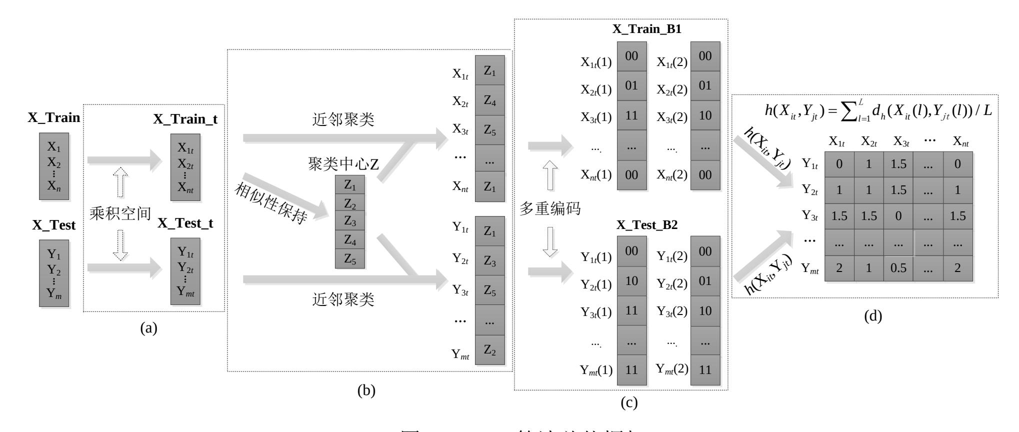


图 3　ISOH 算法总体框架

Fig. 3　Overall framework of ISOH algorithm

1　相似性保持算法

1.1　相似性保持 ISODATA 算法

KMH[14]算法中的聚类中心是随机生成的，导致近邻检索性能不稳定。除此之外，KMH 算法的聚类中心数是定值，在没有先验条件的情况下，由预先给定值得到的聚类效果并不理想。ISODATA 算法[15]根据实际情况动态调整聚类中心数，解决了 KMH 算法聚类中心不可调的问题，但存在初始聚类中心选择盲目性和无法准确设定阈值的问题。为此，本文进行如下改进。

1.1.1 聚类中心初始化

现多数算法常采用最大化原则生成初始聚类中心点[18]，但所生成的聚类结果受第一个初始聚类中心点和比例系数 θ 的影响，并且时间复杂度高，需要通过计算全体样本点与已确定聚类中心点之间的距离来确定新聚类中心点。

为了解决上述问题，本文提出了一种新的初始聚类中心方法，称为最远平均距离法，包含三步：a)将全体样本点的平均值作为第一个聚类中心点，以解决随机性问题；b)选取与第一个聚类中心点之间的距离最远的样本点作为下一个聚类中心点，并依据最近邻原则将样本点划分为两类；c)依次划分样本点数目较多的类，直到满足阈值。伪代码如算法 1 所示。

1.1.2 分裂与合并的阈值

ISODATA 算法[15]通过人机交互修改分裂与合并的阈值，

费时费力。针对以上问题，本文对 ISODATA 算法的阈值进行相应改进。

算法 1 初始化聚类中心

输入：数据集 $X=\{x_1,x_2,\ldots,x_n\}$。

输出：初始聚类中心 $Z=\{z_1,z_2,\ldots,z_K\}$。

1 初始化 $count=\{count[i]|i=1,2,...,K\}$，$K'$；

2 设定中心点为第一个聚类中心 z_1，样本点数目为 $count[1]=n$；

3 重复

3.1 找出 count 中的最大值，并将最大值的索引存入 max；

3.2 在 Z_{max} 所在类中，选取距离聚类中心 Z_{max} 最远的样本点作为第 $K'+1$ 个聚类中心点；

3.3 利用最近邻原则进行归类，更新聚类中心点；

3.4 统计每类样本点数目 count；

3.5 直到 $K'=K$ 为止；

分裂阈值：不同维度上的标准差反映了样本在特征空间不同方向上与聚类中心的位置偏差。如果某个类中样本分散程度较大且样本数量较多，则对其进行分裂操作。若每类分量中标准差的最大值为 σ_{jmax},$j=1,2,\ldots,N_c$，则令 S 为全体 σ_{jamx} 的平均值，分裂阈值 θ_s 为

$$\theta_S=\alpha\times S,\alpha>1 \tag{1}$$

合并阈值：对数据集进行类别划分时，应保证最小化类内距离，最大化类间距离[19]。在类别间最短距离的设定问题上，本文引入了最小生成树[20]，其权重值由各个聚类中心间的距离值表示。假定权重和为 S_w，则合并阈值为

$$\theta_c=\beta\frac{S_w}{N_c-1} \tag{2}$$

其中：$0<\beta<1$；N_c 是聚类中心数。

1.2　相似性保持目标函数

ISODATA 算法在空间量化时会产生量化误差，且采用汉明距离近似代替欧式距离会产生相似性误差。为减小算法误差，若通过罗列所有可能为单元空间分配最优的索引，其时间复杂度高。例如，当编码长度为 b 时，则分配方式有$(2^b!)$种可能。为此，本文交替优化式(3)中的目标函数，同时最小化量化误差和相似性误差。

$$E=E_{quan}+\lambda E_{aff} \tag{3}$$

其中：E_{quan} 表示量化误差，其定义如式(4)所示。其中：$i(x)$表示包含样本点 x 的码字索引；$c_{i(x)}$表示索引为 $i(x)$的码字；T 是样本点数为 n 的训练数据集。

$$E_{quan}=\frac{1}{n}\sum_{x\in T}\|x-c_{i(x)}\|^2 \tag{4}$$

相似性误差是使各个码字分配到的索引可以更近似地表示各码字之间的欧氏距离。相似性误差的公式为

$$E_{aff}=\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}w_{ij}(d(c_i,c_j)-d_h(i,j))^2 \tag{5}$$

其中：$w_{ij}=n_in_j/n^2$；n_i 和 n_j 分别表示索引为 i 和 j 的样本点数；$d(c_i, c_j)$表示码字 c_i 与 c_j 之间的欧氏距离：$d(c_i, c_j)$表示码字 c_i 与 c_j 之间的欧氏距离；$d_h(i,j)$表示索引 i 与 j 之间的汉明距离。

综上，本文迭代优化目标函数(3)的过程如下：

a) 分配步骤，固定码字优化索引。

将每个样本点分配到距离它最近的码字上。

b) 更新步骤，固定索引优化码字。

任何码字的更新取决于所有其他的码字。所以本文顺序优化每个码字 c_i，其他码字固定。

$$c_j=\arg\min c_j(\frac{1}{n}\sum_{x;i(x)=j}\|x-c_j\|^2 \\ +2\lambda\sum_{i;i\neq j}w_{ij}(d(ci,cj)-d_h(i,j))^2) \tag{6}$$

其中：λ 为常量(在本文中 λ=10)；$d_h(i,j)\triangleq v\cdot h^{1/2}(i,j)$表示修正后的索引间汉明距离；$h^{1/2}$ 是汉明距离的平方根；v 是常量，在算法中由主成分哈希算法初始化。

综上，相似性保持算法的基本流程如图 4 所示。

2　编码

2.1　多重编码

由于 ISODATA 算法得到的聚类数目可能会大于给定的聚类数量，而固定长度的二进制编码所能表示二值编码的种类数量是有限的。当编码长度为 b 时,其编码种类只能表示 $K=2^b$ 种。若最终聚类中心数目 K'大于给定值 K，则会超出编码范围。

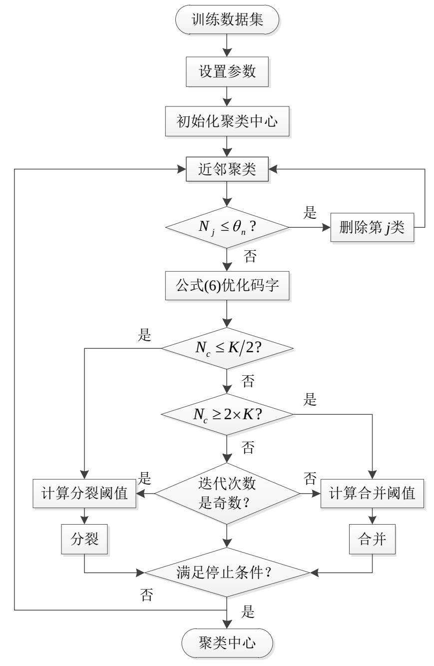


图 4　相似性保持方法流程

Fig. 4　Flow chart of similarity preservation method

若样本点被分成了五个簇，长度为 2 的编码只能表示 4 个聚类中心点。为解决上述问题，本文先将聚类中心点分成 K=2b 类，然后再对其进行编码，如图 5 所示。从图 5 中可以看出，聚类中心点经过聚类后，$c4$ 和 $c5$ 归为一类，映射为相同二值码。这种编码方式可避免聚类中心数超出编码范围，但失去了 ISODATA 算法的聚类优点，而且检索性能较低，复杂度较高。为解决以上问题，本文引入了多重编码[16]，其基本思想是为数据点分配多组二进制编码，并根据平均汉明距离(式(7))检索近邻点。若二进制编码长度为 b，二重编码能表示的数量为 2^{2b}，L 重编码能表示的数量就是 2^{Lb}。

$$d(X_i,Y_j)=\sum_{l=1}^{L}d_h(X_i(l),Y_j(l))/L \tag{7}$$

其中：X_i 表示训练数据集的第 i 个样本点；Y_j 表示测试数据集的第 j 个样本点；$X_i(l)$表示 X_i 的第 l 组编码；$d_h(X_i(l),Y_j(l))$表示 $X_i(l)$与 $Y_j(l)$之间的汉明距离；L 表示为 L 重编码。

如图 6 所示，本文采用二重哈希映射函数对数据集进行编码，为样本点分配两组二进制编码，数据点$(c4, c1)$和$(c4, c5)$之间的平均汉明距离均为 0.5，则在汉明空间内检索 $c4$ 的近邻点时，会同时返回 $c1$ 和 $c5$。

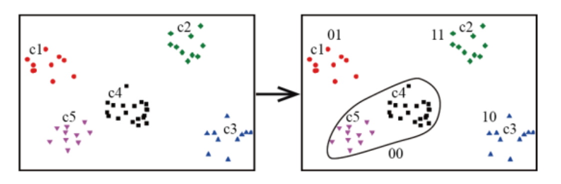


图 5　基于 k-means 聚类的编码示例

Fig. 5　Coding example based on K-means clustering

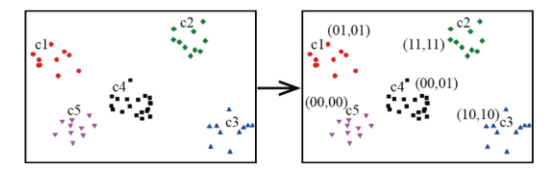


图 6　基于多重哈希函数的编码示例

Fig. 6　Coding examples based on multiple hash functions

2.2　推广到乘积空间

当编码长度 b 很大时，需要计算和存储 2^b 大小的码书，而 ISODATA 算法很难给出太大的码书。为了解决这一问题，在本文中通过将 D 维空间划分为 M 个子空间，再对每个子空间进行相应操作。在划分子空间时，期望划分后的子空间是相互独立的，且每个子空间的方差是均衡的。

在本文中使用主成分分析算法[21]，求前 D 个最大主成分元素，将所有的主成分元素按递减的方式排序，再按从大到小的方式先将 M 个主成分元素分别分配到 M 个桶内，之后依次向特征值总和最小的桶内分配主成分元素，每个桶内最多存放 D/M 个主成分元素，直到所有元素分配完为止。采用上述方式可以将 D 维空间均衡地划分为 M 个子空间。这样 ISOH 算法在编码长度很大的情况下，仍然可以产生很大的码书。

3　实验

3.1　数据集

在本文中，使用的三种公开数据集分别是 SIFT1M[22]、GIST[22]和 CIFAR10[23]。SIFT1M 数据集包含 10^6 个 128 维的训练数据点和 10^4 个查询数据点。在实验中从 10^6 个 128 维的特征点中随机选取 10^4 个数据点做训练数据集，查询数据集是从 10^4 个查询数据集中随机选取 10^3 个。GIST 数据集包含 5×10^5 个训练数据集，10^3 个查询数据集，在实验中从 5×10^5 个训练数据集中随机选取了 10^4 个数据点做训练数据集。CIFAR10 数据库中的数据点是 GIST 特征，它是从 CIFAR10 数据库图像中提取出来的，总共有 6×10^4 个数据点，随机选取 10^3 个数据点做测试数据集，10^4 个数据点做训练数据集。

3.2　评价指标

本文中使用较为广泛的评价标准：召回率(recall)[24]和平均均匀准确率(mean average precision，mAP)[23]。召回率(recall)表示在已经返回的检索结果中，真正近邻数据点索占的比例，其公式为

$$\text{Recall}=\frac{\#(retrieved\ relevant\ points)}{\#(all\ relevant\ points)} \tag{8}$$

其中：#(retrieved relevant points)表示返回结果中真正近邻数据点的数量；#(all relevant points)表示数据集中所有近邻数据点的数量。

平均均匀准确率(mAP)的值反映的是算法返回近邻数据点的速率，其值越大，表示算法返回真正近邻点的速率越快。mAP 值的公式如式（9）所示。

$$mAP=\frac{1}{|Q|}\sum_{i=1}^{Q}\frac{1}{K_i}\frac{j}{rank(j)} \tag{9}$$

其中：$|Q|$表示查询数据集的大小；K_i 表示第 i 个查询数据点的真正近邻点的数量；j 表示查询数据点的第 j 近邻点；$rank(j)$ 返回查询数据点的第 j 近邻点在查询结果中的序号。

3.3　结果与讨论

将 ISOH 算法和现有的一些算法进行比较，所有与 ISOH 算法进行比较的算法的代码都是使用公开默认设置。实验中使用的三种公开数据集分别是 SIFT1M、GIST 和 CIFAR10，选择的测试真值是近邻欧氏距离(NN=10)，测试的编码长度分别是 B=32、64 和 128 位。实验结果如图 7~9 所示。

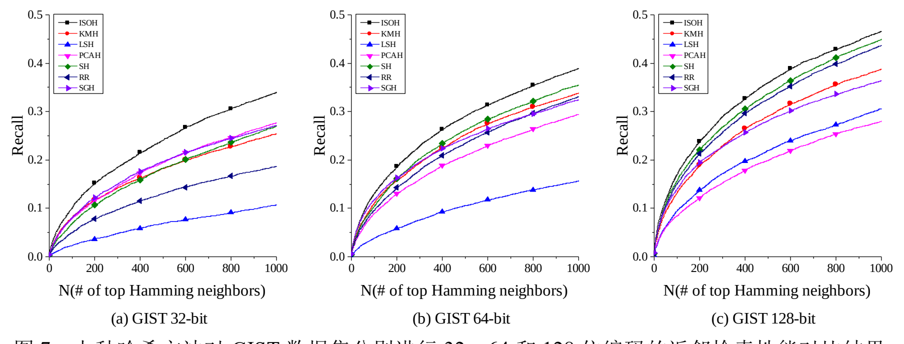


图 7　七种哈希方法对 GIST 数据集分别进行 32、64 和 128 位编码的近邻检索性能对比结果

Fig. 7　ANN search performance of seven hashing methods on GIST dataset encoded using 32, 64, and 128 bit codes

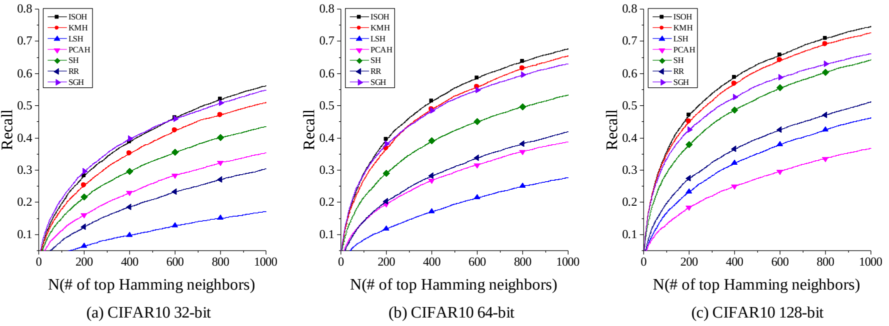
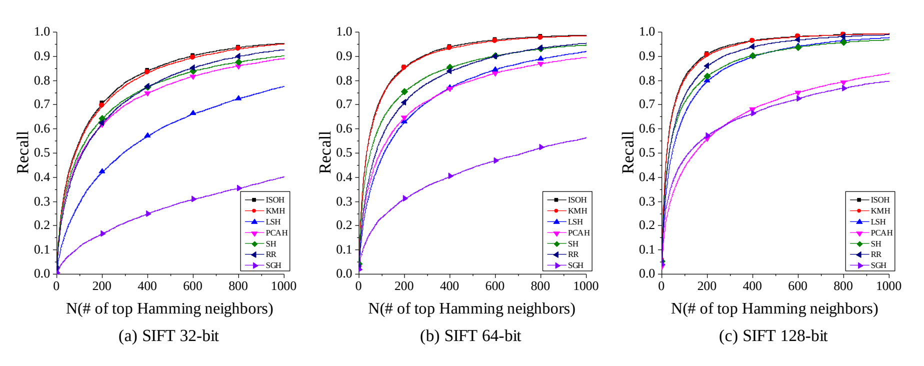


图 8　7 种哈希方法对 CIFAR10 数据集分别进行 32、64 和 128 位编码的近邻检索性能对比结果

Fig. 8　ANN search performance of seven hashing methods on CIFAR10 dataset encoded using 32, 64, and 128 bit codes



(a) SIFT 32-bit　　(b) SIFT 64-bit　　(c) SIFT 128-bit

图 9　7 种哈希方法对 SIFT1M 数据集分别进行 32、64 和 128 位编码的近邻检索性能对比结果

Fig. 9　ANN search performance of seven hashing methods on SIFT1M dataset encoded using 32, 64, and 128 bit codes

从以上实验结果可知，ISOH 算法的检索性能一直处于最优状态。在 SIFT 数据集上，KMH 算法虽然检索性能表现较优，但是从图 7~9 的实验结果中可以看出，ISOH 算法不仅在 SIFT 数据集上的性能一直处于 KMH 算法之上，而且在其他两种公开数据集上其性能也明显高于 KMH 算法；RR 算法在 SIFT 数据集上，编码长度为 128 bit 时，其检索性能良好，但是在其他状态下一直处于较低状态；PCAH 算法在编码长度较小的情况下，其检索性能相对较好，然而当编码长度变大，其检索性能不佳。SH 算法在三种数据集上的检索性能处于中上状态，却始终没有超过 ISOH 算法。在 CIFAR10 数据集上，尽管 SGH 算法检索性能较优，但是与 ISOH 算法相比，其检索性能较弱。LSH 算法在图 9(c)中的检索性能较好，可是在其他情况下检索性能较差。

表 1　各种算法在 GIST 数据库上的 mAP 值/%

Table 1　Map values of each algorithm on GIST datasets /%

bit	32		64		128	
NN	10	100	10	100	10	100
ISOH	1.78	10.40	2.37	12.09	3.54	15.11
KMH	1.21	8.26	2.55	10.53	2.76	11.94
RR	0.99	4.97	1.70	9.36	3.15	14.66
PCAH	1.24	6.63	1.56	7.34	1.81	7.25
LSH	0.39	2.20	0.69	3.60	1.82	8.95
SGH	1.39	7.95	2.05	11.34	3.19	12.99
SH	1.26	6.65	2.05	11.50	3.22	14.96

表 2　各种算法在 CIFAR10 数据库上的 mAP 值/%

Table 2　Map values of each algorithm on CIFAR10 datasets/%

bit	32		64		128	
NN	10	100	10	100	10	100
ISOH	8.54	40.73	13.43	49.42	16.64	49.41
KMH	5.20	30.73	9.64	39.06	12.86	46.05
RR	1.74	12.74	3.76	22.75	5.87	29.25
PCAH	3.66	17.75	4.08	18.40	3.87	17.15
LSH	0.66	6.73	2.28	15.87	5.13	27.34
SGH	8.17	37.58	12.43	48.67	16.43	58.15
SH	5.07	29.89	6.98	35.85	9.40	41.27

表 3　各种算法在 SIFT1M 数据库上的 mAP 值/%

Table 3　Map values of each algorithm on SIFT1M datasets /%

bits	32		64		128	
NN	10	100	10	100	10	100
ISOH	14.99	60.89	26.22	74.51	33.85	81.58
KMH	14.66	60.11	25.56	74.69	33.46	81.28
RR	12.12	53.08	16.10	61.24	26.46	79.56
PCAH	12.28	52.57	14.92	56.27	13.26	48.21
LSH	6.15	31.97	12.14	52.33	22.56	71.08
SGH	1.91	8.84	6.21	20.92	17.80	52.08
SH	13.53	56.34	21.80	69.16	28.58	75.86

从以上近邻检索实验结果可知，ISOH 算法的近邻检索性能优于其他算法。LSH 算法中的哈希映射函数是随机生成，其稳定性差，近邻检索性能偏弱。SH 算法需要假定数据集服从均匀分布，而本文给出的三种数据库都不是均匀分布，其近邻检索性能较弱。SH 算法在构建数据点间的相似图时，其时间复杂度高。为了降低时间复杂度，SGH 算法通过特征转换方式构建相似图，但是为得到较好的检索结果，需要通过人机交互方式不断调整参数 ρ。PCAH 算法将位于映射函数平面两侧的近邻点映射为不同的二进制编码，其汉明距离相应增加，导致近邻检索性能相对较弱。RR 算法通过旋转被特征向量映射后的数据集减小量化误差，可是该算法的旋转矩阵是随机生成的，算法性能稳定性差。KMH 算法使用 K-means 算法进行空间量化，但其预先给定聚类中心数目，适应性差，使得检索性能较弱。

ISOH 算法近邻检索方面优于其他算法可从准确率方面得到直观体现。七种哈希算法在三种不同的数据集上的准确率如表 1~3 所示。从表中可以看出，ISOH 算法的准确率高于其他算法。与 RR 和 PCAH 算法相比，ISOH 算法使用了自适应较强地 ISODATA 算法进行空间量化，允许超立方体进行拉伸，从而使单元空间划分更加细致，算法的准确率显著提高。如表 3 所示，在 SIFT1M 数据集中，编码长度 B=128 和近邻检索真值 NN=100 时，ISOH 算法的准确率高达 81.58%，相较于 PCAH 的 48.21%，整整提高了 33.37%的准确率。在比较算法中，虽然 SH 在三个数据集上的准确率相对较好，但是一直处于中等水平。SGH 算法在 GIST 数据集上的准确率仅次于 ISOH 算法，但是在 SIFT 上的准确率较差。经实验结果证实 ISOH 算法在 CIFAR10、GIST、SIFT1M 等最常用数据集中，不管是在编码长度为 32、64 还是 128，ISOH 算法的准确率相较于 KMH、PCH、RR、LSH、SH 和 SGH 算法等都有明显提高。

ISOH 在聚类中心初始化的问题上，使用最远平均距离方法，避免了初始聚类中心选择上的随机性和盲目性。在现有哈希算法[6,8,10,11,14]中，由于固定编码长度表示种类有限，导致检索精度下降，本文引入了检索精度较高的多重编码。ISOH 算法在不指定数据集的情况下，检索性能也能表现良好。通过各项实验数据的对比，可以清晰地发现 ISOH 算法在各方面具有明显优势。

4　结束语

与 KMH 算法相比，ISOH 算法使用最远平均算法初始化聚类中心，保证了聚类中心选择的可靠性。ISOH 算法可根据样本点的实际情况动态地设定阈值，无须再试探性地修改阈值，降低了算法的训练复杂度。较使用固定编码长度的编码方式，多重编码表示的种类更多，而且算法检索准确率更

高。乘积空间的使用，使算法能以较低代价学习更长的编码。经实验证实，ISOH 算法的近邻检索性能优于 KMH、RR、PCAH、LSH、SH 和 SGH 算法。

参考文献：

[1] Chen Tianlang, Xu Chenliang, Luo Jiebo. Improving text-based person search by spatial matching and adaptive threshold [C]// Proc of IEEE Winter Conference on Applications of Computer Vision. 2018: 1879-1887.

[2] Li Wen, Duan Lixin, Xu Dong, *et al.* Text-based image retrieval using progressive multi-instance learning [C]// Proc of IEEE International Conference on Computer Vision. 2011: 2049-2055.

[3] Chen Shizhi, Yang Xiaodong, Tian Yingli. Discriminative hierarchical K-means tree for large-scale image classification [J]. IEEE Trans on Neural Networks & Learning Systems, 2017, 26 (9): 2200-2205.

[4] Silpaanan C, Hartley R. Optimised KD-trees for fast image descriptor matching [C]// Proc of IEEE Conference on Computer Vision & Pattern Recognition. 2008: 1-8.

[5] Wang Jun, Liu Wei, Kumar S, *et al.* Learning to hash for indexing big data:a survey [J]. Proceedings of the IEEE 2015, 104 (1): 34-57.

[6] Datar M, Immorlica N, Indyk P, *et al.* Locality-sensitive hashing scheme based on p-stable distributions [C]// Proc of the 20th Symposium on Computational Geometry. 2004: 253-262.

[7] Shen Fumin, Zhou Xiang, Yang Yang, *et al.* A fast optimization method for general binary code learning [J]. IEEE Trans on Image Processing, 2016, 25 (12): 5610-5621.

[8] Weiss Y, Torralba A, Fergus R. Spectral hashing [C]// Proc of International Conference on Neural Information Processing Systems. 2008: 1753-1760.

[9] Jiang Qingyuan, Li Wujun. Scalable graph hashing with feature transformation [C]// Proc of International Conference on Artificial Intelligence. 2015: 2248-2254.

[10] Gong Yunchao, Lazebnik S. Iterative quantization: a procrustean approach to learning binary codes [C]// Proc of Computer Vision & Pattern Recognition. 2011: 817-824.

[11] Gong Yunchao, Lazebnik S, Gordo A, *et al.* Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013, 35 (12): 2916-2929.

[12] Jegou H, Douze M, Schmid C, *et al.* Aggregating local descriptors into a compact image representation [J]. Proc CVPR, 2010, 238 (6): 3304-3311.

[13] Irie G, Arai H, Taniguchi Y. Alternating co-quantization for cross-modal hashing [C]// Proc of IEEE International Conference on Computer Vision. 2016: 1886-1894.

[14] He Kaiming, Wen Fang, Sun Jian. K-means hashing: an affinity-preserving quantization method for learning binary compact codes [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2938-2945.

[15] Ball G H, Hall D J. A novel method of data analysis and classification [M]//Data Analysis Machine Learning & Knowledge Discovery.2003.

[16] Xia Yan, He Kaiming, Wen Fang, *et al.* Joint inverted indexing [C]// Proc of IEEE International Conference on Computer Vision. 2013: 3416-3423.

[17] Jegou H, Douze M, Schmid C. Product quantization for nearest neighbor search [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2010, 33 (1): 117-128.

[18] Gu Hongbo, Zhao WanPing. Cluestering algorithm based on max-min distance for students'score analysis in universities and applications [J]. Journal of Hebei University of Engineering, 2010.

[19] He Yan, Ye Qiaolin, Liu Yingan, *et al.* The gepsvm classifier based on L1-norm distance metric [M]. 2016.

[20] Jiang Bo, Zhang Li. Research on minimum spanning tree based on prim algorithm [J]. Computer Engineering & Design, 2009, 30 (13): 3244-3247.

[21] Gupta A, Barbu A. Parameterized principal component analysis [J]. Pattern Recognition, 2018, 78 (6): 215-227.

[22] Jegou H, Douze M, Schmid C. Product quantization for nearest neighbor search. [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2011, 33 (1): 117.

[23] Torralba A, Fergus R, Freeman W T. 80 Million tiny images: a large data set for nonparametric object and scene recognition [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2008, 30(11): 1958-1970.

[24] Fu Xiping, Mccane B, Mills S, *et al.* Nokmeans: non-orthogonal k-means hashing [C]// Proc of Asian Conference on Computer Vision. 2014: 162-177.